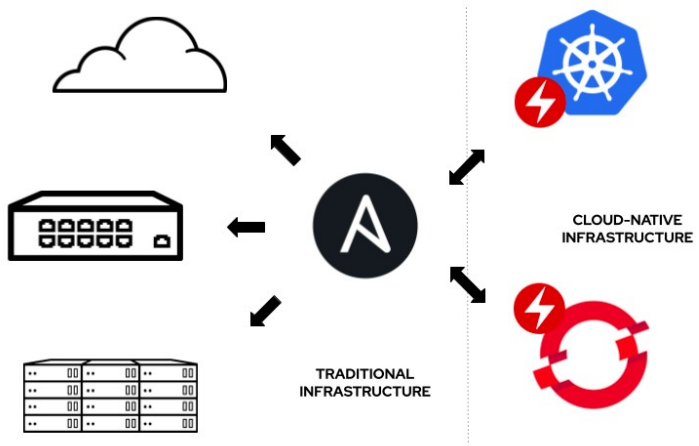
Automate OpenShift with Red Hat Ansible Automation Platform

September 6, 2023 by Rachna Dodia (https://www.ansible.com/blog/author/rachna-dodia)



OpenShift meets Ansible

We have seen many organizations compare themselves on how agile they are in the innovation journeys. When we talk about innovation, there are several aspect d it, like optimizing the existing IT, integrating apps/data/processes, adding anc managing cloud infrastructure or modernizing the applications. But in order to

partner with lines of business to better compete, there is a need to introduce automation.

If someone asks, “What can we automate?” Below is my answer:

"Every non-interruptive CLI command or any UI which exposes a REST-API is an opportunity to automate."

Whether it be orchestrating configurations, deploying applications or managing infrastructure, etc.

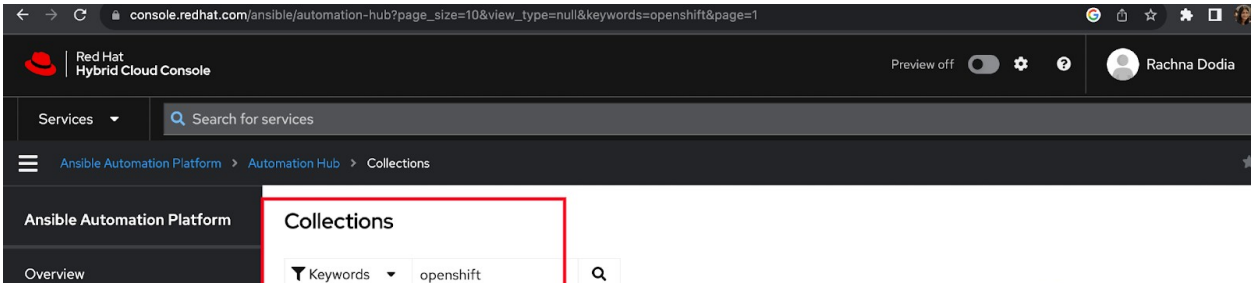
Red Hat Ansible Automation Platform does all that!

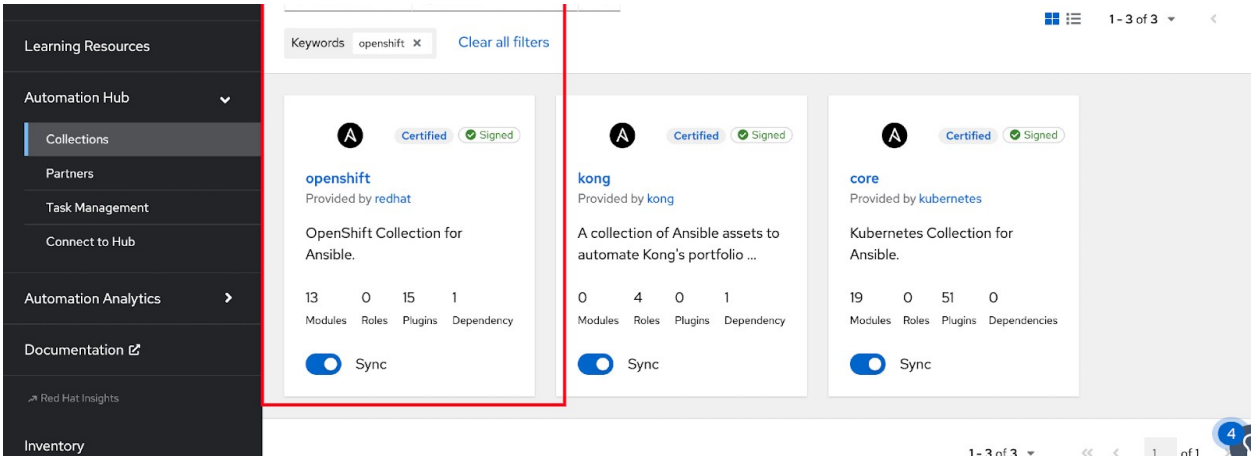
Whether managing cloud or cloud-native systems, there is no difference. In this article, I’ll go through integrating an OpenShift cluster with Ansible Automation Platform to help automating Day 0 and Day 2 activities.

Tying OpenShift with Ansible Automation Platform

When it comes to container orchestration, Red Hat OpenShift has emerged as the leading enterprise level solution, providing a robust platform to manage and scale complex applications with ease. However, manual deployment and management of these applications can be labor intensive and prone to issues.

Enter Ansible Automation Platform: Its strength lies in its simplicity, scalability and ability to bridge the gap between complex IT tasks and efficient execution by turning manual, repetitive tasks into automated workflows. One of its standout components is Red Hat Ansible Certified Content Collections - pre-packed automation content that encompasses modules, plugins, and roles to simplify automation. Among these, the redhat.openshift.k8s collection paves the way for Red Hat OpenShift automation by bringing ease and efficiency of Ansible into your Red Hat OpenShift environments, streamlining your processes.





Red Hat Ansible Certified Content Collections in Ansible automation hub

We have worked with many Banking, Financial Services and Insurance (BFSI) accounts at Red Hat, and we would like to share one of the lessons learned on how Ansible naturally fits into the OpenShift environment.We'll show how the redhat.openshift.k8s collection can not only reduce manual efforts of deployment, but also enables consistency across deployments,efficiency and fewer errors when deploying applications within an OpenShift cluster.

Let us start simple

To start simple, Red Hat OpenShift and Ansible use similar patterns and YAML scripts to describe the desired state of the world:

OPENSHIFT/OC	ANSIBLE

`apiVersion: v1`
`kind: ConfigMap`
`metadata:`
`  name: foo`
`  namespace: default`
`data:`
`  color: red`
	`- name: Create foo config map`
	`  redhat.openshift.k8s:`
	`  definition:`
	`  apiVersion: v1`
	`  kind: ConfigMap`
	`  metadata:`
	`  name: foo`
	`  namespace: default`
	`  data:`
	`  color: “{{ color }}”`

On the left is an OpenShift definition to define any resource. On the right is an Ansible definition to define any OpenShift resource. Notice the similarities?With Ansible, additional features are available so you can maintain the entire definition as a separate Jinja2 template file as in the below example:

```
- name: Create foo config map
  redhat.openshift.k8s:
      template: foo.yml
```

Templatising Openshift Definition

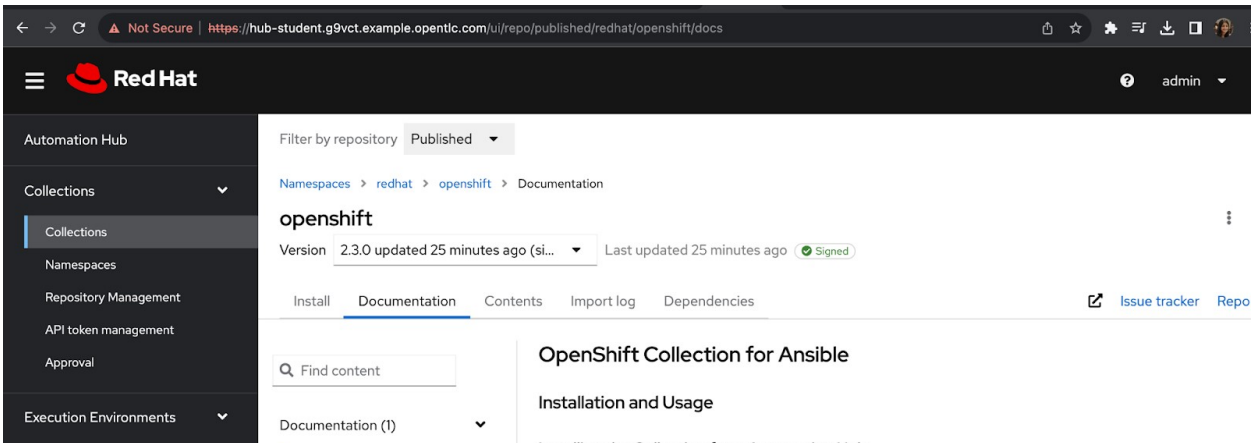
Red Hat OpenShift and Ansible are better together, and one real world example of this is automating the deployment of the Redis leader/follower application on an OpenShift cluster. This scenario takes advantage of the following Ansible Playbook provided in this repository (https://github.com/RachnaDodia/k8_ansible.git).

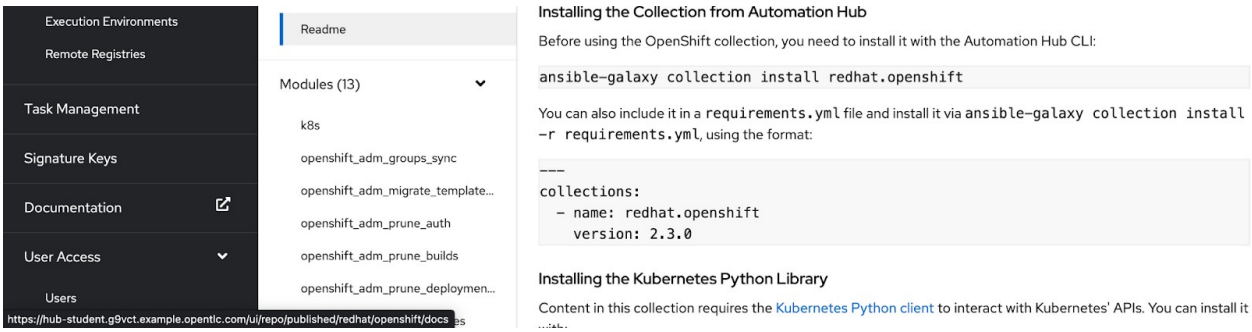
Let us automate!

The solution automates the following:

1. Creation of guestbook namespace, where I'll be carrying out the deployment using the Red Hat Certified Collection
2. Deploying Redis leader, Redis follower and the front end application
3. Configuring networking objects such as services and routes

As a preliminary step, I have downloaded the Red Hat Certified Collection for OpenShift in private automation hub as shown below:





Openshift Collection in Private Automation Hub

I'll configure the below resources on Ansible Automation Platform to get started:

1. *Project to import the Git source code*
   Project Name: Openshift
   Source Control Type: Git
   Source Control URL: https://github.com/RachnaDodia/k8_ansible.git (https://github.com/RachnaDodia/k8_ansible.git)
   Organization: Default <or as per your environment>

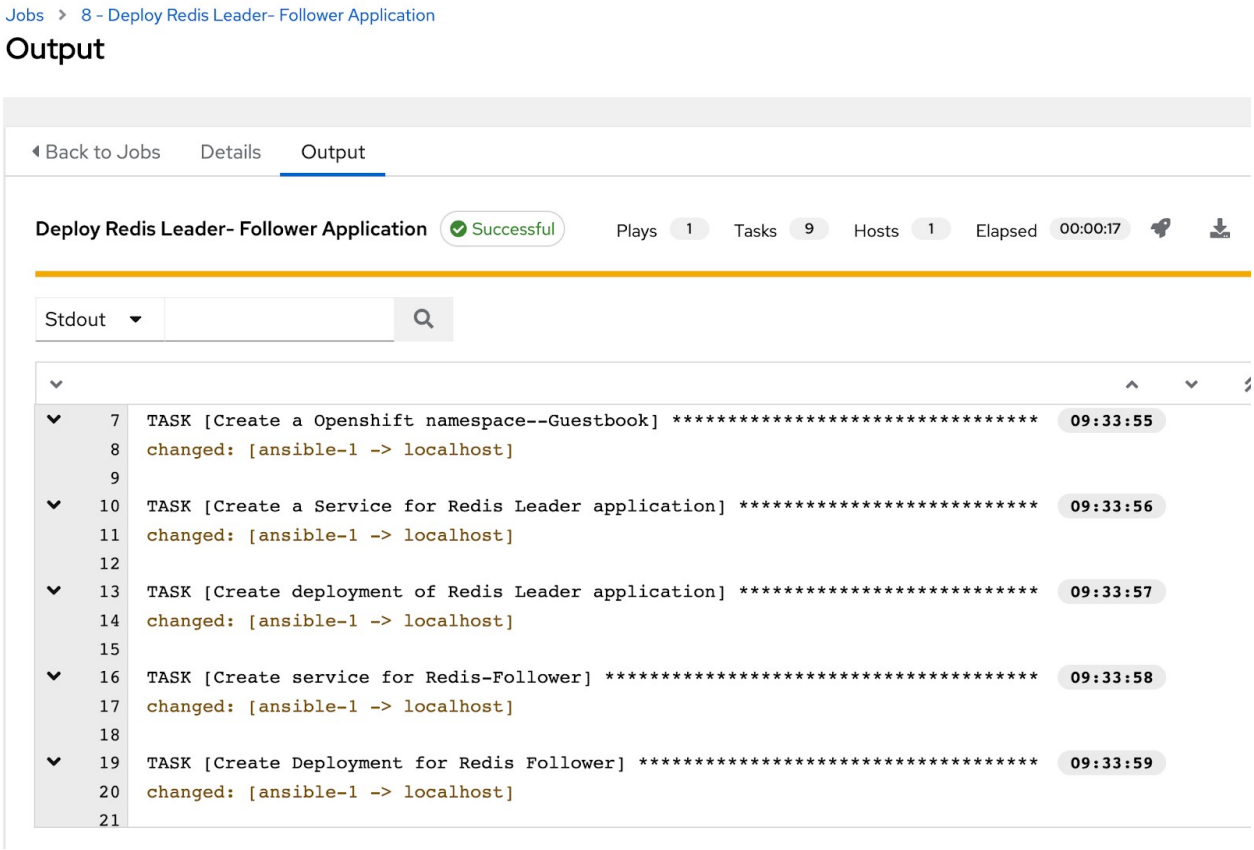
2. *Credentials to store OpenShift API URL and the token for authentication*
   Credential Name: OpenShift
   OpenShift or Kubernetes API Endpoint: <as per your environment>
   API authentication bearer token: <as per your environment>
   Organization: Default <or as per your environment>

3. *Credentials to store private automation hub URL and token for authentication*
   Credential Type: Ansible Galaxy/Ansible automation hub API Token
   Galaxy Server URL: <as per your environment>
   API Token: <as per your environment>
   Organization: Default <or as per your environment>

4. *Template using the resources created above*
   Template Name: Deploy Redis Leader-Follower Application
   Project: OpenShift
   Playbook: deploy.yml
   Credentials: OpenShift
   Execution Environment: Default execution environment

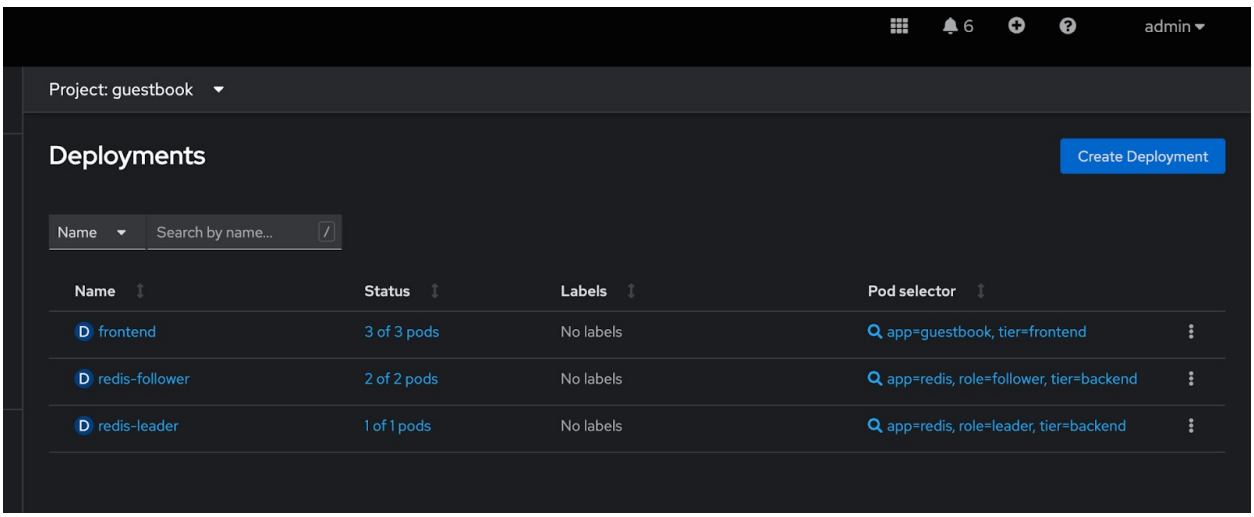
Organization: Default <or as per your environment>

Once the template is configured , **let's** get ready to launch! Here is the output of creating the resources with Ansible Automation Platform:

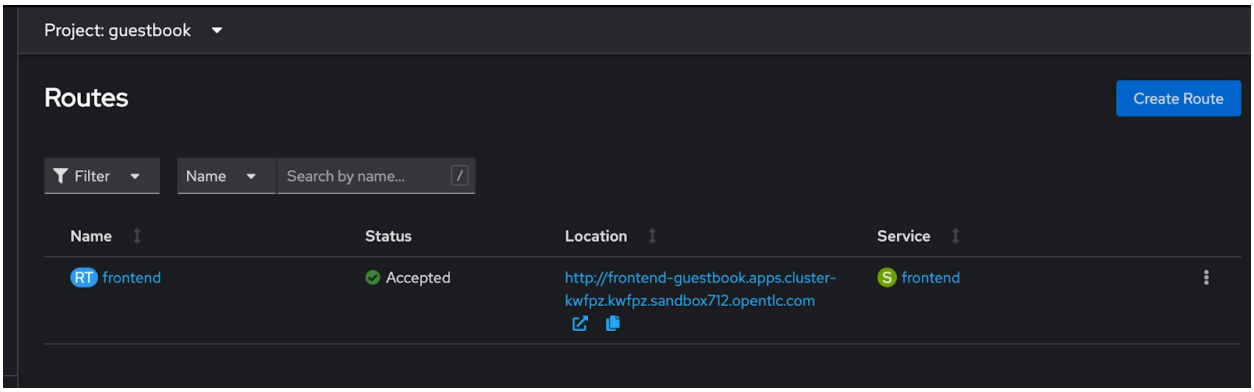


Ansible UI output of successful playbook execution

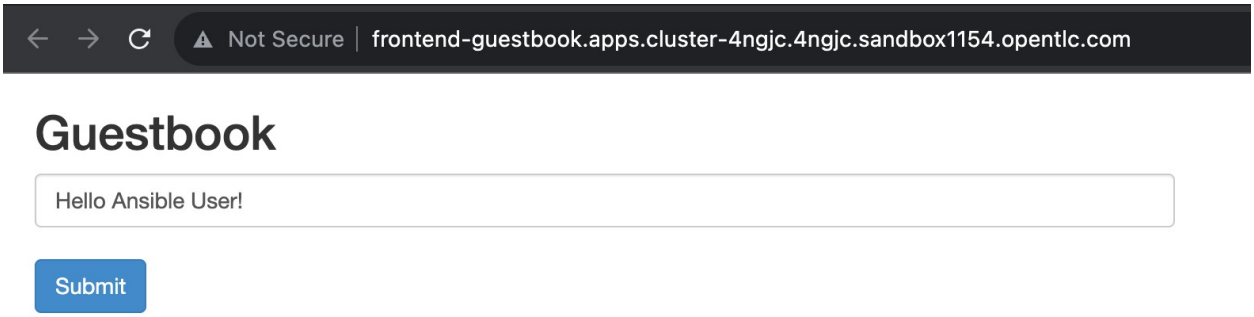
With successful execution of the automation job, let us go to the OpenShift environment to verify the changes:



Deployments in guestbook Namespace



Route mapping to the Service



Front end application

Success! We have deployed the application and created the necessary networking resources.

Takeaways

Thanks to the Red Hat Certified Collection for OpenShift, automating OpenShift is significantly easier. You have seen now how effortlessly multiple configurations can be performed on an OpenShift cluster. Not only can you begin with Day 0 operations, but you can manage Day 1 and Day 2 operations as well. This can be improved upon by maintaining all the Configuration as Code (CaC), where you would separate configuration settings from the actual code. Ideally, you can store that configuration data in source control such as Git, and easily run and tweak it to match different environments.

A key piece of advice: Don't limit yourself as an OpenShift admin, but take it to the

next level with automation.

Where to go next

- Get hands-on with on-demand Ansible Automation Platform self-paced exercises (https://www.redhat.com/en/engage/redhat-ansible-automation-202108061218) - We have a variety of interactive in-browser exercises to experience Ansible Automation Platform in action.
- Trial subscription (http://red.ht/try_ansible) - Are you ready to install on-premises? Get your own trial subscription for unlimited access to all the components of Ansible Automation Platform.
- Red Hat Developer subscription (https://developers.redhat.com/about) - Did you know you can get a free Red Hat Developer subscription to learn in your home lab. Register and get access to all the latest tools and technologies that Red Hat offers.
- Subscribe (https://www.youtube.com/ansibleautomation) to the Red Hat Ansible Automation Platform YouTube channel
- Follow Red Hat Ansible Twitter (https://twitter.com/ansible) - Do you have questions or an automation project you want to show off? Tweet at us!

Share:

Topics:
OpenShift (https://www.ansible.com/blog/topic/openshift), Ansible Automation Platform (https://www.ansible.com/blog/topic/ansible-automation-platform)

Rachna Dodia (https://www.ansible.com/blog/author/rachna-dodia)

Rachna is a Specialist Solution Architect at Red Hat having key expertise in automation and cloud. She has an overall experience of 5 years in the IT industry ,where she started her career as a Application Support Engineer for Global Contact Centers at Barclays Technology Center. As a Red Hatter, she has worked with multiple BFSI customers and helped them in their digital transformation journey.

Search this site on Google

Search

Categories

- Ansible content collections (https://www.ansible.com/blog/topic/ansible-content-collections)
- Automation controller (https://www.ansible.com/blog/topic/automation-controller)
- Automation execution environments (https://www.ansible.com/blog/topic/automation-execution-environments)
- Automation hub (https://www.ansible.com/blog/topic/automation-hub)
- Automation mesh (https://www.ansible.com/blog/topic/automation-mesh)
- Cloud Automation (https://www.ansible.com/blog/topic/cloud-automation)
- Configuration Management (https://www.ansible.com/blog/topic/configuration-management)

- Network Automation (https://www.ansible.com/blog/topic/network-automation)
- Security Automation (https://www.ansible.com/blog/topic/security-automation)
- Windows (https://www.ansible.com/blog/topic/windows)

See All

(//www.ansible.com/blog/rss.xml) RSS Feed (//www.ansible.com/blog/rss.xml)

Red Hat Ansible

Ansible is an open source community project sponsored by Red Hat, it's the simplest way to automate IT. Ansible is the only automation language that can be used across entire IT teams from systems and network administrators to developers and managers.

About Us (https://www.ansible.com/about)

Our Story (https://www.ansible.com/about)

Press Center (https://www.ansible.com/press-center)

Careers (https://www.redhat.com/en/jobs)

(https://twitter.com/ansible) (https://www.facebook.com/ansibleautomation)

(https://www.linkedin.com/company/3545/) (https://github.com/ansible/ansible)

- QUICK LINKS
- PRODUCTS
- USE CASES
- INTEGRATIONS
- RESOURCES
- COMMUNITY
- RELATED SITES

Red Hat Summit (https://www.redhat.com/summit)